Modeling critical mechatronic systems with Petri Nets and feared scenarios derivation

Hamid DEMMOU¹, Edwige GUILHEM², Sarhane KHALFAOUI¹,², Robert VALETTE¹

¹ LAAS CNRS, Toulouse
{hamid, robert}@laas.fr

² PSA Peugeot Citroën DINQ/SIPP/EIEV, La Garenne Colombes
{guilhem1, khalfaou}@mpsa.com

Abstract. The aim of this work is to propose a logical based approach for deriving the critical scenarios from a Petri Net model of the mechatronic system. This model represents both nominal (normal) behavior and the behavior under failures. First, we present the case study: the physical system, its configurations, its continuous dynamics, its failures and the repair actions. Second, we detail the Petri Net model in a modular way starting with a model of the nominal operation. Then, a model of the behavior under failures is presented and finally these two partial models are merged into the complete one. Third, the logical based approach for deriving the critical scenarios is detailed and illustrated on one feared behavior of the system.

Keywords. Petri Nets, mechatronic, safety

1 Introduction

Mechatronic systems combine electrical, mechanical, hydraulic and electronic technologies associated with a computer based control. They are used in many applications such as defense, nuclear or aerospace industries and in the automotive as well. The benefit of such systems lies in the very large flexibility thanks to the software implementation of the control functions. Consequently, functions improving safety can be easily added. During design, it is important to characterize feared behaviors which are critical. As they have to be rare (security constraints), simulation is typically insufficient because only nominal behaviors are explored [1]. The aim of this work is to propose a logical based approach for deriving the critical scenarios from a Petri Net model of the mechatronic system. This model represents both nominal (normal) behavior and the behavior under failures.
First, we present the case study: the physical system, its configurations, its continuous dynamics, its failures and the repair actions. Second, we detail the Petri Net model in a modular way starting with a model of the nominal operation. Then, a model of the behavior under failures is presented and finally these two partial models are merged into the complete one. Third, the logical based approach for deriving the critical scenarios is detailed and illustrated on one feared behavior of the system.

2 A case study

Our case study is the « conjunction-disjunction system ». Its aim is to supply hydraulic systems (as the brake system, the hydraulic suspension and the gearbox) with a sufficient pressure. It controls the oil pressure in an accumulator in order to maintain it within a given interval. It is composed of a pump, a cut-out electrical valve, an accumulator, a sensor and a computer.

![Diagram of the conjunction-disjunction system](image)

Fig. 1. The conjunction-disjunction system

2.1 Description of components

- The pump is considered as perfect and gives an oil flow of $Q_{pump} = 3 l / mn$.
- The sensor measures the accumulator pressure.
- There are two warning lights, one for the high pressure warning light and the other for the low pressure one.
- The cut-out electrical valve is an intermediate device between the pump and the hydraulic systems which need oil under pressure.
- The accumulator is made up of a sphere divided into two rooms by a flexible membrane, a closed one containing nitrogen, and the other open on the oil circuit. It imposes a pressure depending on
the input and the output flow of oil (the pump’s flow and the consumers one respectively). It works in the following way:

Initially, the accumulator is full of gas, and we have $P_0 = 62$ bar and $V_0 = 400$ cm$^3$. If a volume $V$ of liquid is injected into the accumulator with no thermal transfer, the pressure becomes:

$$P = P_0^\gamma (V_0/(V_0-V))$$

with $\gamma = 1.4$.

![Fig. 2. The accumulator](image)

- The role of the computer is to control the cut-out electrical valve. It implements the following three rules with $P_{\text{min}} = 145$ bar, $P_{\text{max}} = 175$ bar, $P_{\text{alarm min}} = 120$ bar and $P_{\text{alarm max}} = 200$ bar.
  - If $P \leq P_{\text{min}}$, then the pump supplies the accumulator and the consumers (conjunction phase) (rule 1)
  - If $P \geq P_{\text{max}}$, then the pump is shunted to the tank (disjunction phase) (rule 2)
  - If $P < P_{\text{alarm min}}$ or $P > P_{\text{alarm max}}$, then an alarm is triggered (rule 3)

- We modeled the hydraulic systems by a two level flow consumer. The high level represents an oil consumption corresponding, for example, to a braking phase. The low one (here considered to be zero) corresponds to a situation in which the hydraulic systems are not solicited, for instance when one drives at a constant speed on a flat road.

### 2.2 Dynamic behavior

At the beginning, the accumulator is empty, $P = P_0$ and the electrical valve is closed. The sensor measures the current level of pressure $P$ in the accumulator, and the computer opens the electrical valve, it is the conjunction phase (1) : the pressure $P$ is increasing:

$$P = P_0^\gamma (V_0/(V_0-V))$$

and

$$V = \int Q_e dt - \int Q_{\text{consumers}} dt$$

where $V$ is the volume, $Q_e$ the input oil flow and $Q_{\text{consumers}}$ the output one.
When \( P > P_{\text{max}} \), the computer closes the electrical valve. Therefore, the pump is connected to the tank and the accumulator supplies alone the consumers: it is the disjunction phase (2): the pressure is decreasing: \( P = P_0*(V_0/(V_0-V))^\gamma \) and \( V = -\int Q_{\text{consumers}} \, dt \).

When \( P < P_{\text{min}} \) a new conjunction phase begins, and so on...

One can notice that the dynamic behavior of the system can be summarized by four Algebraic Differential Equations (two for the conjunction phase (1) and two for the disjunction one (2)) according to the values of the input flow \( Q_e \) and of the output one \( Q_{\text{consumers}} \):

\[
\begin{align*}
P &= P_0*(V_0/(V_0-V))^\gamma \\
V &= \int Q_e \, dt - \int Q_{\text{consumers}} \, dt \quad \text{with} \quad Q_e = \begin{cases} 0 \Rightarrow (2) \\
Q_{\text{pump}} = 3 \, l/\text{mn} \Rightarrow (1) \end{cases} \\
Q_{\text{consumers}} &= \begin{cases} 0 \Rightarrow (1,2) \\
Q_e = 2 \, l/\text{mn} \Rightarrow (1,2) \end{cases}.
\end{align*}
\]

### 2.3 Failures and repair actions

The electrical valve can be blocked in open or closed position. When it is blocked, the following repair action is executed: it is shaken during 0.1 s, if it is released (probability of success \( p \)) the system recovers its nominal operation. In case of failure (probability \( 1-p \)), the system drifts until one of the two warning lights is turned on.

### 3 The model

As it has been explained in the introduction, the first step of our approach consists in modeling the system. It is an hybrid system and we need a clear separation between the discrete part and the continuous one in order to drive the feared scenarios. The model is therefore based on Differential Predicate Transitions Nets as defined in [2].

#### 3.1 Nominal operation

The places \( P_1, P_2, P_3 \) and \( P_4 \) (see figure 3) represent the different configurations of the system. The places \( P_1 \) and \( P_4 \) represent the conjunction phase. \( P_1 \) denotes the configuration in which \( Q_{\text{consumers}} = Q_e \). \( P_2 \) represents the configuration in which \( Q_{\text{consumers}} = 0 \). The place \( P_2 \) and \( P_3 \) are modeling the disjunction phase when consumers are whether asking or not for oil.

These four places correspond to a \( p \)-invariant. The equations presented in section 2.2. are attached to these places.
The place $P_5$ represents the electrical valve when it is open, while $P_{15}$ models it when it is closed. Transitions $T_{12}$ and $T_{43}$ are modeling the computer’s order to close the electrical valve. They are fired when the condition $P \geq P_{\text{max}}$ becomes true. Transition $T_{21}$ represents the opening of the valve when the condition $P \leq P_{\text{min}}$ becomes true. The transitions $T_{14}$, $T_{41}$, $T_{23}$ and $T_{32}$ are modeling the consumption flow change (high and low consumption level).

**Fig. 3.** The nominal working model

### 3.2 Failure and repair model of the cut-out electrical valve

The figure 4 depicts the way we modeled the repair action of the electrical valve when it fails while it is open and while the system is providing the hydraulic systems with oil, otherwise when the place $P_5$ is marked. We proceed the same way in the other cases.

The failure of the electrical valve is represented by putting a new token into the place $P_0$. If the electrical valve is open when this token is available and if $P \geq P_{\text{max}}$, then we fire the transition $T_5$ and marks the place $P_6$ which represents the blockage of the electrical valve in open position. Therefore, the repair starts by firing immediately the transition $T_8$ and marking the output place $P_7$. The repair will succeed with a probability $p$. If it is successful then we fire the transition $T_7$ and mark $P_8$ immediately. In this case, the electrical valve is opened successfully and the token in $P_8$ returns to $P_5$ by firing $T_9$. Otherwise, if it fails the transition $T_8$ will be fired and the place $P_9$ marked.
3.3 The alarm triggering model

The alarm is triggered when the oil pressure exceeds the limits allowed. As in figure 5, thanks to a guard (for example \( P > P_{\text{alarm}_{\text{max}}} \)) associated with the transition \( T_1 \), one can supervise the pressure level in the place \( P_1 \). If the condition \( P > P_{\text{alarm}_{\text{max}}} \) is true, then \( T_1 \) is fired and the place \( P_h \) is marked.

3.4 The complete model

Now, we will describe how we have modeled the whole system using Petri Nets (see figure 6).

The nominal operation model is completed in one hand by the models of failure and repair of the electrical valve given above, and in the other hand by the warning light model (see figures 3,4,5).

Concerning the failure and repair model, it has to be duplicated in order to cover all the situations in which the electrical valve has to be opened or closed (transitions \( T_{12}, T_{21}, \) and \( T_{43} \)).

Let us consider the case when the valve is blocked in open position (impossibility of transition from conjunction to disjunction phase). In this case, the transition \( T_5 \) is modeling the possibility of the valve failure. In figure 6, \( T_5 \) must be connected with place \( P_1 \) by a loop because it is the case of disjunction with low consumption. If there is a failure (a token in \( P_0 \)), there is a conflict between \( T_5 \) and \( T_{12} \). We have to assume that transition \( T_5 \) has a higher priority than \( T_{12} \).

We have to point out the fact that for the sake of clarity places \( P_0, P_5 \) and \( P_{15} \) have been represented twice in the right and left part of figure 6.
4 A method for deriving the feared events

4.1 Principle of the approach

The approach for deriving the feared events and feared scenarios consists of four steps.

We first execute a Monte-Carlo simulation of the model of the system starting from its initial state. As a result, this simulation will give the marking probability of these places. The nominal states will correspond to a higher probability. As we stated in paragraph 3, we have a hybrid system. The two Petri Net simulators (Design/CPN and Miss-RdP) are able to simulate such systems. The use of Design/CPN [1] [3] implies a discretization of the continuous part (the equations given in section 2.2.). The use of Miss RdP implies a linearization of the continuous part.

The second step consists in the determination of the critical states, typically it will be sink places and their interpretations (the state they denote) will be clear. In addition, they will generally never be marked during the short Monte-Carlo simulation of the first step.

The third step is a set of backward reasonings starting from the sink places obtained at the preceding step. The framework of the reasoning is the linear logic [4]. The backward chaining will stop when all the places containing tokens correspond to nominal states.
The fourth step consists in constructing all the possible scenarios starting from the terminal states of the backward reasonings done at the preceding step. All the conflict situations between the nominal scenarios and the feared ones are identified.

The benefit of this approach is that it allows us, for each feared scenario, to clearly define the last situation in which it separates from the nominal ones.

4.2 Formal framework for backward reasoning and scenario derivation

The third and fourth step can obviously be based on the reachability graph of the Petri Net representing the system. This approach has been studied by [1]. The drawbacks are the combinatorial explosion of the state space and the interleaving semantics which will merge the firing of the significant transitions with the firings of transitions corresponding to other state changes occurring in parallel in an other part of the system.

Our approach exploits the equivalence between the reachability in Petri Nets and the provability of specific sequents in linear logic. Linear logic has been defined by J.Y. Girard [4] in order to deal with resources and state changes. It is based on sequent calculus. A sequent such as $A, B \vdash C, D$ means that from the hypotheses A and B, the conclusions C or D can be derived. For us, a sequent will denote a scenario that is an initial partial marking, a list of transition firings and a final partial marking.

More precisely, linear logic is a restriction of classical logic in the sense that, in order to be consistent with the notions of resources, the weakening and the contraction rules of the classical sequent calculus (proof that a sequent is syntactically correct) have been eliminated. As a consequence, the two connectives “and” and “or” have been replaced by four connectives. One of them is used in our approach: the connective $\otimes$ which denotes the conjunction of resources. For example, $A \otimes A \otimes B$ denotes a marking such that two tokens are in place A and one token in place B. In addition, we use the modal connective $!$ which denotes a non defined number of resources. For instance, $!T_1$ means that we consider a scenario in which transition $T_1$ can be fired an undefined number of times (zero included).

By using the canonical proof tree [5], we introduce no spurious partial order, which means no spurious causal relations between states. This means that during the third step only the places which are involved in a critical scenario leading to a feared state are examined. During the fourth step, only the
places which are a logical consequence of the initial partial marking are derived. The events which can concurrently occur during the scenarios are not considered.

4.3 Example

Let us go back to the conjunction-disjunction system and its model in figure 6. The simulation step points out the fact that $P_1$, $P_2$, $P_3$ and $P_4$ are nominal states. The second step identifies the two feared partial states: places $P_h$ and $P_b$. They respectively represent the high and low pressure alarms.

To illustrate the third and the fourth step we only consider the case of low pressure alarm. This means that the backward reasoning starts with place $P_b$. The starting sequent is: $M_1, !(Ti) \vdash P_b \otimes M_2$ where $M_1$ represents the initial partial marking, $T_i$ all the transitions of the net and $M_2$ the minimal context strictly necessary for the backward reasoning. At the end of this step, we derive the following sequent: $P_2, T_2 \vdash P_b$.

The fourth step generates all the scenarios starting with a partial marking including $P_2$. As $T_2$, $T_{23}$, $T_{21}$ and $T_{15}$ are output transitions of $P_2$, four conflicting scenarios are first generated.

- The first one is: $P_2, T_2 \vdash P_b$ (Sequent 1).
- The second sequent is: $P_2, !(T_{23},T_{32}) \vdash P_2$ (Sequent 2),
- The third one is $P_2 \otimes P_{15}, !(T_{21},T_{12}) \vdash P_2 \otimes P_{15}$ (Sequent 3),
- During the generation of the fourth one, a conflict appears between transitions $T_{17}$ and $T_{18}$ ($T_{17}$ represents that the repair action of the valve failed). This leads to the two following sequents:

  $$P_2 \otimes P_{15} \otimes P_0, T_{15},T_{16},T_{18},T_{19} \vdash P_2 \otimes P_{15}$$ (Sequent 4),
  $$P_2 \otimes P_{15} \otimes P_0, T_{15},T_{16},T_{17},T_2 \vdash P_b \otimes P_{19}$$ (Sequent 5).

Sequent 1 is exactly the one derived by the backward reasoning. Sequent 2 characterizes the nominal consumption changes, possible in the disjunction phase. When deriving sequent 3, we had to enrich the context of the partial initial marking by adding one token in $P_{15}$ (the valve is closed). It denotes the nominal transition between conjunction and disjunction with a high consumption. For sequents 4 and 5, we have to enrich the context of the partial initial marking by adding one token in $P_{15}$ and one in $P_0$ (occurrence of the valve failure). Sequent 4 characterizes a behavior such that the failure of the valve
occurs and is taken into account (transition $T_{15}$) and is successfully repaired (transition $T_{18}$). Finally, sequent 5 is the one which completely describes the critical scenario leading to the feared partial state. It complements sequent 1 with a context describing the valve state.

By analyzing the conflicts between the scenarios, we point out the following facts. The separation between nominal operation and feared scenario results from a series of two conflicts: the first one between transitions $T_{15}$ and $T_{21}$ when a failure occurs, and the second one between transitions $T_{17}$ and $T_{18}$ when the repair action fails.

5 Conclusions

In this paper, we have illustrated the fact that it is possible to logically derive critical feared scenarios from a Petri Net model of the behavior of a mechatronic system. It is a preliminary study and in the approach some issues remain open. They will be addressed in future developments. In step 3, the stop criteria (a nominal state is reached) must guarantee that the backward procedure goes far enough to derive rich information without generating the entire reachability graph. In a similar way, the initial partial marking of the scenarios have to be rich enough to capture the critical behavior, but we have to avoid generating unnecessary transition firings. Finally, our approach has to be complemented by a quantitative evaluation of scenario durations and probabilities.

6 References


